



Technical Newsletter

This Newsletter No. GN26-0898
Date September 30, 1977

Base Publication No. GC26-3819-3
File No. S370-30

Prerequisite Newsletters None

OS/VS Virtual Storage Access Method (VSAM) Options for Advanced Applications

© Copyright IBM Corp. 1973, 1975, 1976

This technical newsletter, a part of Release 6 of OS/VS1 and Release 3.7 of OS/VS2, provides replacement pages for the subject publication. These replacement pages remain in effect for any subsequent releases unless specifically altered. Pages to be removed and inserted are:

Cover, edition notice
3, 4
7, 8
19-20.1 (20.1 added)
71, 72

Each technical change is marked by a vertical line to the left of the change.

Summary of Amendments

Control Blocks in Common (CBIC) option (MVS only), which allows a user to place the VSAM control blocks associated with a VSAM data set into the Common Service Area (CSA) of MVS.

Note: Please file this cover letter at the back of the base publication to provide a record of changes.



GC26-3819-3
File No. S370-30

Systems

**OS/VS Virtual Storage
Access Method (VSAM)
Options for
Advanced Applications**

VS1 Release 6
VS2 Release 3.7

PLEASE RETURN TO:
PANSOPHIC SYSTEM INC.
DATACENTER LIBRARY

IBM

Fourth Edition (January 1976)

This edition, as amended by technical newsletter GN26-0898, applies both to Release 6 of OS/VS1 and to Release 3.7 of OS/VS2, and to any subsequent releases of either system unless otherwise indicated in new editions or technical newsletters.

Significant system changes are summarized under "Summary of Amendments" following the list of figures. Each technical change is marked by a vertical line to the left of the change.

Information in this publication is subject to significant change. Any such changes will be published in new editions or technical newsletters. Before using the publication, consult the latest *IBM System/370 Bibliography*, GC20-0001, and the technical newsletters that amend the bibliography, to learn which editions and technical newsletters are applicable and current.

Requests for copies of IBM publications should be made to the IBM branch office that serves you.

Forms for readers' comments are provided at the back of the publication. If the forms have been removed, comments may be addressed to IBM Corporation, P.O. Box 50020, Programming Publishing, San Jose, California 95150. All comments and suggestions become the property of IBM.

HOW TO USE THIS PUBLICATION

This publication describes applications of VSAM (Virtual Storage Access Method) that might be effective in the development of utility programs and system control programs.

The chapters of this publication are:

- “Summary of Amendments,” which gives the changes to this publication for the releases of OS/VS1 and OS/VS2.
- “Processing Control Intervals,” which describes how to gain access to the whole contents of a control interval (rather than to individual data records) and how to manage your own I/O (input/output) buffers with control-interval access.
- “Sharing Resources among Data Sets,” which describes how to build a VSAM resource pool for sharing I/O buffers, I/O-related control blocks, and channel programs and how to manage buffers in a buffer pool.
- “Processing a VSAM Index,” which gives the format of a record in the index of a key-sequenced data set and explains how to gain access to an index using either the GETIX and PUTIX macros or the GET and PUT macros.
- “Processing a VSAM Catalog,” which describes how to gain access to a VSAM catalog as a catalog, by way of the SHOWCAT macro, and as a data set, by way of the GET and PUT macros.
- “Building Parameter Lists for GENCB, MODCB, SHOWCB, and TESTCB,” which gives the format of the parameter lists used by GENCB, MODCB, SHOWCB, and TESTCB and explains how to build a parameter list to generate, modify, display, or test an access-method control block, exit list, or request parameter list.
- “Appendix: Operand Notation for the Macros,” which describes list and execute forms of the macros presented in this publication and expressions that can be used to code the operands of each macro.

This publication also has:

- “Glossary,” which defines technical terms used in this publication.
- “Index,” which identifies the pages on which the subjects of this publication are discussed.

The formats of the control information in a control interval, of a VSAM index record, and of the macros for building parameter lists for GENCB, MODCB, SHOWCB, and TESTCB are subject to change. The publication of this information implies no commitment on the part of IBM to maintain these formats in future releases.

Conventions Used in the Publication

The conventions used in this publication for presenting the formats of macros indicate whether an operand is optional, how to specify the value for an operand, and how to punctuate a macro. The conventions are:

- Expressions enclosed in brackets, [], are optional.
- Items separated by an OR sign, |, are alternatives, only one of which may be specified. The alternatives are enclosed in braces, { }.
- An underlined **item** is the default when you don't specify anything for an operand.
- Ellipses, ..., indicate either that you may repeat the preceding item or that other items, not relevant to the discussion, are omitted.
- Capitalized **BOLD** expressions, parentheses, commas, and equal signs must be entered as shown, except that, unless otherwise noted, parentheses aren't required if you specify only one item.
- Lowercase *italic* expressions are variables for which you may specify one of a number of expressions.

Required Publications

The *OS/VS Virtual Storage Access Method (VSAM) Programmer's Guide*, GC26-3838, is recommended for learning about VSAM data sets and the macros you use to gain access to them. Rather than repeat information from the *VSAM Programmer's Guide*, this publication assumes you are familiar with the *VSAM Programmer's Guide*. This publication discusses the operands of some of the macros described in that publication and some of the return codes from those macros, but it relies on that publication for a full presentation of the macros and return codes.

Related Publications

The following publications are referred to for additional information about particular subjects:

OS/VS Publication

- *OS/VS1 Checkpoint/Restart*, GC26-3876, and *OS/VS2 MVS Checkpoint/Restart*, GC26-3877, for information about the use of checkpoint/restart with a VSAM data set.

OS/VS1 Publications

- *OS/VS1 Access Method Services*, GC26-3840, for information about the commands used to process a catalog as a catalog.
- *OS/VS1 Virtual Storage Access Method (VSAM) Logic*, SY26-3841, for information about the format and arrangement of records in a VSAM catalog.

OS/VS2 Publications

- *OS/VS2 Access Method Services*, GC26-3841, for information about the commands used to process a catalog as a catalog.
- *OS/VS2 Catalog Management Logic*, SY26-3826, for information about the format and arrangement of records in a VSAM catalog.

CONTENTS

How to Use This Publication	3
Conventions Used in the Publication.....	4
Required Publications.....	4
Related Publications.....	4
Figures	9
Summary of Amendments	11
OS/VS1 Release 5 and OS/VS2 Release 3.7.....	11
GENCB, MODCB, SHOWCB, and TESTCB.....	11
OS/VS1 Release 4 and OS/VS2 Independent Component.....	11
Control-Interval Access.....	11
Sharing Resources.....	11
Index Processing.....	11
Catalog Processing.....	11
OS/VS2 Release 3.....	11
Processing Control Intervals	13
The Format of a Control Interval.....	13
CIDF—Control-Interval Definition Field.....	13
RDF—Record Definition Field.....	14
How to Gain Access to a Control Interval.....	17
Managing Your Own I/O Buffers.....	18
Improved Control-Interval Access.....	18
CBIC (Control Blocks in Common) Option (MVS only).....	20
Error Return Codes.....	20
Sharing Resources Among Data Sets	21
Providing a Resource Pool.....	22
Deciding How Big a Resource Pool to Provide.....	22
BLDVRP: Building a Resource Pool.....	23
OPEN: Connecting a Data Set to a Resource Pool.....	25
DLVRP: Deleting a Resource Pool.....	26
Managing I/O Buffers.....	27
Deferring Write Requests.....	27
Relating Deferred Requests by Transaction ID.....	28
WRTBFR: Writing Buffers Whose Writing Has Been Deferred.....	28
Handling Exits to Physical-Error-Analysis Routines.....	30
Using the JRNAD Exit with Shared Resources.....	31
Preventing Deadlock in Exclusive Control.....	32
Using Control-Interval Access with Shared Resources.....	33
Summary of Restrictions for Shared Resources.....	37
Processing the Index of a Key-Sequenced Data Set	39
The Format of an Index Record.....	39
The Header Portion.....	40
The Free-Control-Interval-Entry Portion.....	40
The Index-Entry Portion.....	42
How to Gain Access to a Key-Sequenced Data Set's Index.....	46
Opening the Cluster.....	46
Opening the Index Component of the Cluster.....	46

Processing a VSAM Catalog	47
Gaining Access to a Catalog as a Catalog	47
Using the SHOWCAT Macro	47
Gaining Access to a Catalog as a Data Set	51
The Format of a VSAM Catalog	51
Opening a Catalog	52
Gaining Access to a Catalog	53
Building Parameter Lists for GENCB, MODCB, SHOWCB, and TESTCB	55
The Format of the Parameter Lists	55
Building Header and Element Entries	57
Passing Control Directly to VSAM	59
Modifying and Displaying the RECLLEN Field of an RPL Directly	60
Error Return Codes	61
Appendix: Operand Notation for the Macros	63
List and Execute Forms	63
The BLDVRP Macro	63
The DLVRP Macro	64
The SHOWCAT Macro	64
Expressions That Can Be Used for Operands	66
Glossary	69
Index	71

Its CPU (central processing unit) processing is faster than that of normal control-interval access. In OS/VS2 a processing program can achieve the best performance with improved control-interval access by combining it with SRB dispatching. (SRB dispatching is described in *OS/VS2 System Programming Library: Supervisor*.)

To open an object for improved control-interval access, the named object must:

- Be an entry-sequenced or relative record cluster; the data component of a key-sequenced, entry-sequenced, or relative record cluster; or the index component of a key-sequenced cluster (index records must not be replicated).
- Have control intervals the same size as physical records. When you use Access Method Services DEFINE to define the object, you can specify control-interval size equal to a physical-record size used for the device on which the object is stored. VSAM uses physical-record sizes of 512, 1024, 2048, and 4096 bytes. (4096 is not used, however, for the IBM 2314 Direct Access Storage Facility.)
- Not be empty.

If these restrictions aren't met, the object can't be opened.

To process a data set with improved control-interval access, a request must be:

- Defined by a single RPL (no chained request parameter lists). VSAM ignores the NXTRPL operand.
- A direct GET, GET for update, or PUT for update (no POINT, no processing empty data sets—a relative record data set with slots formatted is considered not to be empty, even if no slot contains a record).
- Synchronous (no CHECK, no ENDREQ).

With improved control-interval access, VSAM assumes (without checking) that an RPL whose ACB has MACFR=ICI has OPTCD=(CNV, DIR, SYN) that a PUT is for update (OPTCD=UPD), and that your buffer length (specified in RPL AREALEN=*number*) is correct. Since VSAM does not check these operands, you should debug your program with ACB MACRF=NCI, then change to ICI.

With improved control-interval access, VSAM doesn't take JRNAD exits and doesn't keep statistics (which are normally available by way of SHOWCB from the access-method control block).

With improved control-interval access, you may specify that control blocks are to be fixed in real storage (ACB MACRF=(CFX,...)). If you so specify, your I/O buffers must also be fixed in real storage. Having your control blocks fixed in real storage, but not your I/O buffers, may cause physical errors or unpredictable results. If you specify MACRF=CFX without ICI, VSAM ignores CFX. NFX is the default—it indicates that buffers are not fixed in real storage, except for an I/O operation. A program must be authorized to fix pages in real storage—either in supervisor state with protection key 0 or 7, or link-edited with authorization. (The authorized program facility is described in *OS/VS2 System Programming Library: Supervisor*.) An unauthorized request is ignored.

CBIC (Control Blocks in Common) Option (MVS Only)

The CBIC option places the VSAM control blocks associated with a VSAM data set into the common service area (CSA) of MVS. When you open the VSAM data set with the CBIC option, you must be in supervisor state with a protection key from 0 to 7. If this is not the case, the open will fail. The control block structure and VSAM I/O support are essentially the same with CBIC as without except for the location of the control block structure. The user-related control blocks are generated in the protection key of the user (from 0 to 7). The system-related control blocks are generated in key 0. The VSAM control block structure generated with the CBIC option retains normal control block structure connections to the region which opened the VSAM data set (such as the DEB chained to the region's TCB, etc.).

The CBIC option is invoked when a VSAM data set is opened. To invoke this option, you must set the CBIC in the ACB to one prior to OPEN. The CBIC bit is located at offset X'33' (ACBINFL2) in the ACB, bit 2 (ACBCBIC).

The CBIC option may be used only with the ICI option. The following restrictions apply when using the CBIC option:

- CBIC cannot be used with LSR or GSR specified.
- The catalog, user, or system CRA, VVIC (Virtual Volume Inventory Control), swap, or system data sets cannot be used with the CBIC option.
- A VSAM checkpoint is not allowed if CBIC data sets are being used in a region.

If another region accesses the data set's control block structure in the CSA through VSAM record management, the following conditions should be observed:

- An open must only be issued against the data set from the first region.
- The ACB of the user who opened the data set with the CBIC option must be used.
- CLOSE and CLOSE (TYPE=T) cannot be issued for the data set (only the user who opened the data set with the CBIC option can close the data set).
- A region must have the same storage protect key as the user who opened the data set with the CBIC option.
- User exit routines should be accessible from all regions accessing the data set with the CBIC option.

Error Return Codes

Some of the error return codes described in the *VSAM Programmer's Guide* have other meanings for control-interval access.

Return Codes from OPEN

One return code from OPEN in the ERROR field in the ACB has additional meanings for control-interval access:

ERROR

Code	Meaning
160(A0)	The operands specified in the ACB (or GENCB) macro are inconsistent with each other or with information in the catalog. Inconsistencies include: <ul style="list-style-type: none">• MACRF=ICI or UBF is specified along with LSR or GSR.• MACRF=ICI is specified, but control-interval size for the component doesn't equal physical-record size or the object to be opened is a key-sequenced cluster or an index component with record replication.
224(E0)	MACRF=ICI is specified, but the data set is empty.

Logical-Error Return Codes

Some logical-error return codes have additional meanings for control-interval access, and some apply only to control-interval access:

FDBK

Code	Meaning
76(4C)	A key-sequenced or relative record data set cannot be extended by way of control-interval access.
80(50)	ERASE is invalid for control-interval access.
84(54)	OPTCD=LOC is inconsistent with MACRF=UBF.
104(68)	The RPL options are either invalid or inconsistent: <ul style="list-style-type: none">• OPTCD=CNV is specified along with BWD.• With MACRF=ICI, OPTCD=(CNV, DIR, SYN) must be specified.• With MACRF=ICI, PUT with OPTCD=NUP is invalid.• With MACRF=ICI, POINT, CHECK and ENDREQ are invalid.
200(C8)	Control-interval access is not allowed for processing by way of a path.

Physical-Error Return Codes

For physical errors, the RBA field in the request parameter list gives the relative byte address of the control interval in relation to which the physical error occurred.

INDEX

For additional information about any subject listed in this index, refer to the publications that are listed under the same subject in either *OS/VS1 Master Index*, GC24-5104, or *OS/VS2 Master Index*, GC28-0693.

A

- ACB macro
 - deferring writes from a buffer pool 27
 - processing an index 46
 - using a resource pool 25
 - using control-interval access 17-18
- access to
 - catalog as a catalog 47-51
 - catalog as a data set 51-53
 - contents of control intervals 17-19,33
 - index records 46
- Access Method Services
 - processing a catalog as a catalog 47
 - VERIFY command 52
- access-method control block
 - displaying statistics about a buffer pool 23
 - identifying a catalog with the SHOWCAT macro 49
- address list, in parameter lists of GENCB, MODCB, SHOWCB, and TESTCB macros 55
- addressed access
 - intermixed with control-interval access 17
 - to a catalog 53
 - to an index 46
- ALTER command
 - processing a catalog as a catalog 47
- alternate indexes
 - processing as a data set 39
 - providing buffers for shared resources 24
 - upgrading when processing base cluster with control-interval access 33
- authorized program facility
 - fixing pages in real storage 19,24
 - opening a VS2 catalog 52

B

- base RBA, in index-record header
 - how used to calculate the RBA of a control interval 41
 - of control area 41
 - of index 41
- BLDVRP macro
 - building a VSAM resources pool 23-25
 - execute form 64
 - list form 63
- boundary values, of keys, through rear compression 43
- buffer modification, journalling of 31-32
- buffer pool
 - connecting a data set to 25
 - deferring writes from 27
 - deleting 26
 - managing 27-36
 - providing 22-25
- buffer, marking for output 35-36
- buffer, releasing from control 35-36
- buffering, I/O
 - using work area as a buffer 18-19
 - with shared resources 27-36

(see also buffer pool)

- building a resource pool 23-25
- building parameter lists for GENCB, MODCB, SHOWCB, and TESTCB macros
 - coding example 57-58
 - format of parameter lists 55-56
 - macros used 57

C

- cancelling I/O-error processing in a JRNAD exit routine 31-32
- catalog entries, relationship among 47-48,51-52
- catalog, VSAM
 - format of 51-52
 - order of search with SHOWCAT macro 48
 - processing as a catalog 47-51
 - processing as a data set 51-53
- CBIC (Control Blocks in Common) 20
- chaining request parameter lists
 - control-interval access 17
 - not allowed with
 - improved control-interval access 19
 - MRKBFR macro 36
 - SCHBFR macro 34
 - WRTBFR macro 29
 - CHECK macro
 - with control-interval access 18
 - with the WRTBFR macro 29
- checkpoint routine
 - restriction with a global resource pool 37
 - writing buffers 29
- CIDF (control information definition field)
 - end of data 13-14
 - format 13-14
 - illustration 13,15,16
- closing a data set
 - writing buffers 29
- codes
 - control information in control interval 13-14,39
 - index record 41-42,44-45
 - JRNAD exit parameter list 31-32
 - parameter lists of GENCB, MODCB, SHOWCB, and TESTCB macros 57
 - (see also error return codes)
- coding example, of building parameter lists for GENCB, MODCB, SHOWCB, and TESTCB macros 57-58
- common area of virtual storage 22,25
- compression, key
 - control information in index entry 42-45
 - examples 42-45
 - front 42-43
 - highest possible key value 44-45
 - illustration 44
 - rear 43
 - boundary values 43
- control area
 - base RBA of, in a sequence-set index record 41
 - none in an index 39
- control information
 - control interval
 - CIDF 13-14
 - RDF 14-16
 - responsibility for maintaining, with control-interval access 17

control information (continued)

- index record
 - free-control-interval entry 40
 - header 40-41
 - index entry 42-45
 - section displacement field 42
- parameter lists of GENCB, MODCB, SHOWCB, and TESTCB macros
 - address list 55
 - element entry 55-56
 - header entry 55-56
- control interval
 - control information
 - CIDF 13-14
 - RDF 14-16
 - responsibility for maintaining, with control-interval access 17
 - illustration 13,15,16
 - in a catalog 51
 - in a key-sequenced data set
 - free, governed by free-control-interval entry in index 40
 - used, governed by index entry in index 40-41
 - in an index 39
 - password 17
- control interval definition field (CIDF)
 - end of data 13-14
 - format 13-14
 - illustration 13,15,16
- control-area splits, journalling of 31-32
- control-interval access
 - improved 18-19
 - debugging with normal control-interval access 19
 - passwords 17
 - responsibility for maintaining control information 17
 - to a data set 17-19,33
 - to an index 46

D

- data records, position in a control interval 14-16
- deadlock in exclusive control, preventing 32
- deferring write requests 27-28
- DEFINE command
 - processing a catalog as a catalog 47
- DELETE command
 - processing a catalog as a catalog 47
- deleting a resource pool 26-27
- direct access, to control intervals 17
- displaying information about an unopened data set (SHOWCAT macro) 47-51
- displaying statistics about a buffer pool 23
- displaying transaction ID of request parameter list 28
- distributed free space 40
- DLVRP macro
 - deleting a resource pool 26-27
 - execute form 64
- DSNAME parameter, in JCL, for specifying an index 46

E

- element entry, in parameter lists of GENCY, MODCB, SHOWCB, and TESTCB macros 55-56
- illustration 56
- using macro to build 57-58
 - coding example 57-58
- elimination of characters from key (*see* key compression)
- end of data
 - indicated by CIDF 13-14
 - verifying 52-53
- ENDREQ macro
 - giving up exclusive control 32
 - like MRKBFR macro 35
 - used with WRTBFR macro 29
- entry
 - catalog 47-48,51-52
 - element, in parameter lists of GENCB, MODCB, SHOWCB, and TESTCB macros 55-56
 - free-control-interval, in index record 40
 - header, in parameter lists of GENCB, MODCB, SHOWCB, and TESTCB macros 55-56
 - index 41-45
- entry-sequenced data set
 - loading with control-interval access 18
 - processing with shared resources 33
 - RDFs 14-16
- EODAD exit with control-interval access 17
- ERASE macro, not used with control-interval access 18
- error return codes
 - from BLDVRP macro 25
 - from DLVRP macro 27
 - from GENCB, MODCB, SHOWCB, and TESTCB macros, with user-built parameter lists 61
 - from GETIX macro 46
 - from MRKBFR macro 36
 - from OPEN macro
 - catalog 52
 - control-interval access 20
 - shared resources 26
 - from PUTIX macro 46
 - from request macros (GET, PUT, Etc.)
 - buffers not available 29
 - for control-interval access 20
 - physical errors
 - control-interval access 20
 - JRNAD exit 31-32
 - program check with global shared resources 25
 - from SCHBFR macro 35
 - from SHOWCAT macro 51
 - from SHOWCB macro 23,28
 - from TESTCB macro 28
 - from WRTBFR macro 30
- event dispatching (improved control-interval access) 19
- EXCEPTION exit with deferred writes 30
- exclusive control
 - preventing deadlocks 32
 - releasing 35-36
- execute forms of macros 63-65
- expressions that can be used for operands 66-67

OS/VS Virtual Storage Access Method (VSAM)
Options for Advanced Applications
GC26-3819-3

**Reader's
Comment
Form**

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. This form may be used to communicate your views about this publication. They will be sent to the author's department for whatever review and action, if any, is deemed appropriate. Comments may be written in your own language; use of English is not required.

IBM shall have the nonexclusive right, in its discretion, to use and distribute all submitted information, in any form, for any and all purposes, without obligation of any kind to the submitter. Your interest is appreciated.

Note: Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.

Note: Staples can cause problems with automated mail sorting equipment.
Please use pressure sensitive or other gummed tape to seal this form.

Reader's Comment Form

Fold and Tape

First Class Permit
Number 6090
San Jose, California

Business Reply Mail

No postage necessary if mailed in the U.S.A.

Postage will be paid by:

**IBM Corporation
P.O. Box 50020
Programming Publishing
San Jose, California 95150**

Fold and Tape



International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, N.Y. 10604

IBM World Trade Americas/Far East Corporation
Town of Mount Pleasant, Route 9, North Tarrytown, N.Y., U.S.A. 10591

IBM World Trade Europe/Middle East/Africa Corporation
360 Hamilton Avenue, White Plains, N.Y., U.S.A. 10601